ML Algorithms Explained:

Supervised ML algorithms:

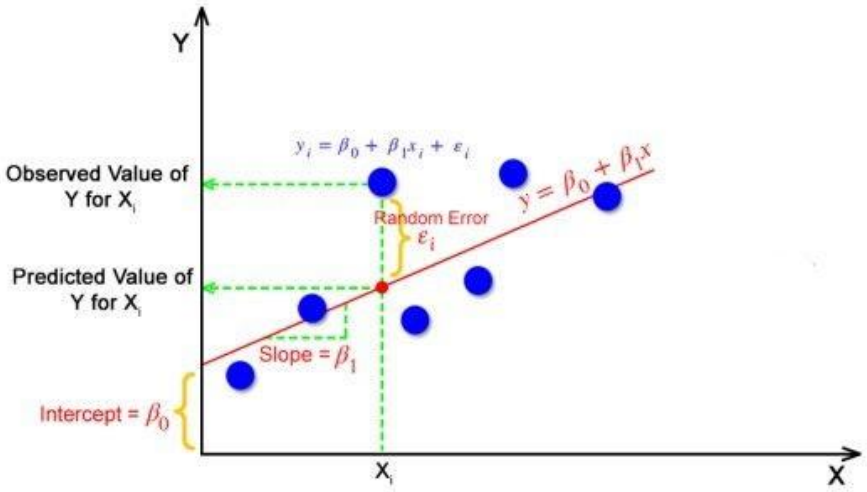
1. Linear Regression
2. KNN
3. Naïve Bayes
4. Decision trees
5. Random Forest
6. Support Vector Machines
7. Neural Networks

Unsupervised ML algorithms:

1. K Means Clustering
2. Association rules

==========

1. Linear Regression: (https://www.youtube.com/watch?v=CtsRRUddV2s)



Helps to solve supervised ml problems.

Y = mx + c for single feature

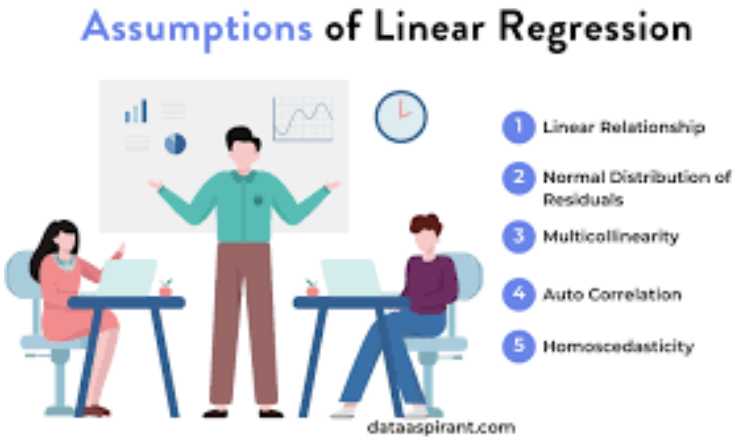
Where m is slope and c is the intercept of the fitted line.

Y = b0 + b1x + b2x + ...bnx (for b0 to bn features)

Used Case: Regression ex: House Price Prediction

Assumptions of Linear Regression: (https://www.youtube.com/watch?v=EmSNAtcHLm8)

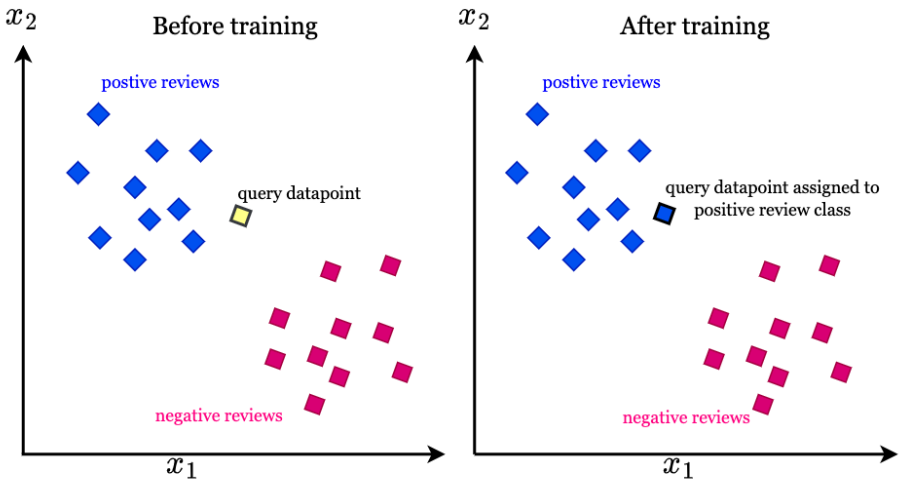
1. Linear Relationship between dependant and independent variables.
2. No Multicollinearity or No co-relation between independent variables.
3. Normal Residuals. (residual error should be in normal distribution I.e. mean 0 std dev. 1 if plotted graphically)
4. Homoscedasticity (Homo: same/uniform, scedasticity: Spread/scatter) residual spreads should be uniform.
5. No Auto Correlation of errors.



Limitations:

- Sensitive to outliers.
- Sensitive to missing values.

2. KNN: (Birds of a feather flock together)



Link: https://www.youtube.com/watch?v=IPqZKn_cMts&t=12s

Assumption: KNN Algorithm assumes that similar things exist in close proximity.

Use Case: Both Classification and Regression

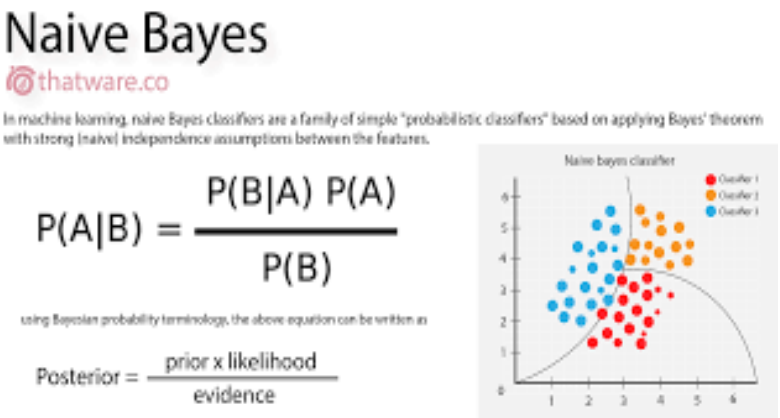
Explanation:

- Load data from dataset.
- Initialize 'k' to your chosen numbers of neighbors.
- For each sample of data, calculate the distance between the query (new unknown sample) and current example of data.
- Add distance and index of data in ordered collection
- Sort collection on basis of distances (ascending order)
- Pick 1st 'k' entries from sorted collection
- In classification case, get mode of 'k' labels
- In regression, get median of the labels.
- Distance can be calculated by:
    - Euclidian distance
    - Manhattan distance

Limitations:

- Sensitive to outliers.
- Sensitive to missing values
- Not good for huge dataset or a greater number of features.

3. Naïve Bayes: (https://www.youtube.com/watch?v=xXeoWE4KmmY)



Use Case: Classification

Assumption: This algorithm assumes that all features are independent and hence the word naïve.

Formula: P(A) = P(B). P(B|A) / P(A|B)

Proving Bayes theorem:

For **dependent** events, joint probability of A and B (such as deck of cards or picking green color marbles from box of yellow and green marbles)

P (A and B) = P(A). P(A|B)

Joint probability is commutative:

P (A and B) = P (B and A)

P(A). P(A|B) = P(B). P(B|A)

Thus, P(A) = P(B). P(B|A) / P(A|B)

Pros:

- Used for high dimensional data such as text detection or spam detection
- Assumption is all features to be independent, makes this algorithm very fast.

Cons:

- Less accurate due to its assumption and not real case condition

4. Decision Trees: (https://youtu.be/ynTCUngbFHA?si=0ao-A_HSrQTEZVR3)

Use case: Both for classification and regression.

Purity check test:

Entropy

Gini Impurity

Root node selection test: Information Gain

Pruning in decision trees